

## **REMARKS**

The application now contains claims 1-16, 18, 19, 21-27, 29-35, 37-41. Claims 17, 20, 28, 36 have been cancelled. In view of the foregoing amendments and following remarks, Applicants respectfully request allowance of the application.

## **THE SECTION 112 AMENDMENTS HAVE BEEN OVERCOME.**

Claims 2-5, 7-10 and 36 stand rejected as indefinite. Applicants have overcome the outstanding rejections by amendment. Specifically, with respect to claims 2-5 and 7-10, Applicants have adopted the Examiner's language suggestion.

## **THE CLAIMS DEFINE OVER THE ART**

Claims 1, 6, 11-19, 23, 28 and 30-36 stand rejected as anticipated by Black, The Block-Based Trace Cache.

Claim 6 now recites:

determining, from flags identifying locations from which the instructions in the instruction segment were fetched, whether the instruction segment satisfies a filtering condition, and

Black does not teach or suggest this subject matter. Black merely states that a fill unit detects cached blocks and does not attempt to insert them into a block cache a second time. Meaning, If the block is already present in the cache, there is no need to overwrite the block with the exact same data. Black makes no mention of any flags used by the system. He certainly does not teach or suggest flags that identify locations from which instructions in the instruction segment were fetched. Accordingly, claim 6 defines over the cited art.

Claims 11 and 18 recite:

a segment builder provided in communication with the instruction cache system, to store a new instruction segment in the segment cache after the instruction segment has been built a predetermined plural number of times.

Black does not teach this subject matter either. In Black, the fill unit discards a block if the block is already present in the block cache. This is different than a segment builder that stores

a new instruction segment after the segment has been built a predetermined number of times. Claims 11 and 18, therefore, define over the cited art.

Dependent claims 12 and 19 further recite a history map to identify instruction segments that have been built previously but were discarded. Black teaches no such features. In Black, the fill unit discards a block only after it determines that the block is already stored in the block cache. Therefore, claims 12 and 19 also define over the art.

Claim 23 recites:

determining whether the instruction segment satisfies a filtering condition based on source locations of instructions within the instruction segment, and

Black does not teach or suggest this subject matter. As noted, Black merely prevents duplicative overwriting within his block cache. He does not base any filtering condition upon the source locations of instructions within an instruction segment. Claim 23, therefore, defines over the art.

Claim 30 recites:

a segment cache to store instruction segments after the respective instruction segments have been built at least twice.

Black does not teach such subject matter. As noted, he merely discloses to prevent operations in a block cache when a block would overwrite itself. Claim 30, therefore, defines over the art.

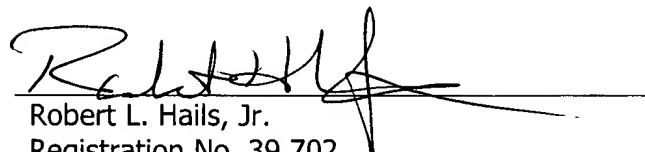
#### **NEW CLAIMS 37-41**

New claims 37-41 are submitted for examination. These claims recite that instruction segments are built and discarded until an instruction segment has been built at least twice, in which case, it is stored to a segment cache. Black also fails to teach or suggest such subject matter.

Allowance is solicited.

Respectfully submitted,

Date: April 7, 2004

  
Robert L. Hails, Jr.  
Registration No. 39,702  
(Attorney for Intel Corporation)

KENYON & KENYON  
1500 K Street, N.W.  
Washington, D.C. 20005  
Ph.: (202) 220-4200  
Fax.: (202) 220-42